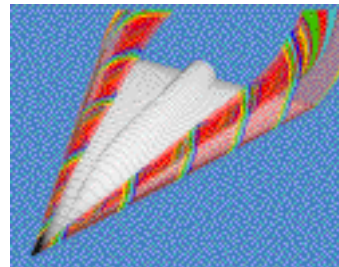


Special Section: Unsteady Visualization



- [Understanding the Basic Challenges of Visualizing Unsteady Computational Fluid Dynamics Data](#)
- [Wind Tunnel Smoke Simulated in Numerical Flow Visualization](#)
- [Improved Techniques for Unsteady Visualization Analysis](#)
- [pV3 Combines Flow Solver, Parallel Interactive Viewing, Feature Extraction](#)
- [Unsteady Flow Technique Reveals New Phenomena](#)



[Next Article](#)[Contents](#)[Main Menu](#)[NAS Home](#)

Understanding the Basic Challenges of Visualizing Unsteady Computational Fluid Dynamics Data

by Kristina D. Miceli and [Michael Gerald-Yamasaki](#)

Advances in high-performance computing have led to an increase in the size and complexity of CFD [computational fluid dynamics] simulations. One aspect of this increased complexity is the time-varying, or "unsteady," nature of the simulation. Visualization technology, which has become an essential tool in analyzing time-invariant, or "steady-state," CFD simulations, must address the additional challenges posed by unsteady CFD simulations.

Incorporating the Time Variable

Early CFD research focused only on solving the physical equations for steady state fluid flow. This restriction was necessary due to the complexity of the algorithms and simulation software and the limitations of computer hardware. With advances in high-performance computing capabilities and simulation technology, scientists can now model fluid flow as it changes over time. Unsteady flow (also referred to as "time-dependent") simulations are important because they provide scientists with a more realistic depiction of a fluid flow, which is necessary when designing dynamic vehicles, such as airplanes.

Time Variable Poses Challenges

Techniques for visualizing fluid flow have been developed in conjunction with simulation technology. Techniques for steady-state flows are readily available. Examples of these techniques, which are used to view the physical variables computed in a simulation, include contouring and color coding for scalar data (for example, pressure) and streamlines for vector data (for example, velocity).

If a simulation is unsteady, visualization techniques must be used to capture the time-varying characteristics of flow. Typically, this involves processing datasets orders of magnitude larger than steady-state simulations and applying more sophisticated visualization algorithms.

Two Approaches

When visualizing an unsteady flow simulation, two approaches may be taken. In the first, steady flow visualization techniques are applied to each timestep (a discrete instant in time) of data in the simulation and viewed sequentially as an animation; in the second, the time variable is explicitly included in computing the visualization technique.

The first approach involves using a steady visualization technique, such as color coding, and applying it to each timestep in a simulation. The time variable isn't used when computing each visualization, but is applied when animating the [individual visualization frames](#). The challenge here is the management of large datasets, which represent potentially thousands of timesteps.

The second approach is more complicated, requiring algorithms that incorporate the time variable when computing the visualization. These complex algorithms are computationally intensive, challenging visualization researchers to optimize their computations -- as well as manage the large datasets -- for maximum performance.

One unsteady visualization technique that incorporates the time variable is unsteady particle tracing, a technique that depicts the path of particles in a flow as they move over time. [See ["Improved Techniques for Unsteady Visualization Analysis."](#)] Unsteady particle traces are in the same family of techniques as streamlines, which create instantaneous representations of particle paths in a steady-state flow field.

The algorithms required to compute unsteady particle traces are much more complex than those for steady-state flows. They must account for factors such as changes in the computational grids over time, additional interpolation requirements across timesteps, and numerical integration complications.

`Time' Affects Mode of Operation

Dataset size and visualization algorithm complexity have direct influence on the time required to compute and render a visualization. In turn, the time to generate the visualization, affects the "mode of operation" in which the scientists explore their data. These modes, ranked by the time required to produce a visualization, can be defined as real-time, interactive, or batch.

Basic visualization techniques applied to steady-state data can often be computed almost instantaneously, allowing scientists to manipulate and probe their data with immediate response, or in "real time." Because of the larger datasets and more complicated algorithms in unsteady visualization, real-time visualization is often very difficult to achieve. The mode of operation may degrade to interactive, where the computation and rendering of a visualization might take up to several seconds but still falls within the range that is reasonable for a data analysis session. Finally, the batch mode of operation is used for computations that are lengthy due to the size of the dataset and algorithm complexity.

The mode of operation has a significant impact on how a visualization system is designed and how it is used by scientists. Currently, most unsteady visualizations are run in interactive and batch modes. As

technology advances, the goal of visualization researchers is to work toward the development of real time visualization tools.

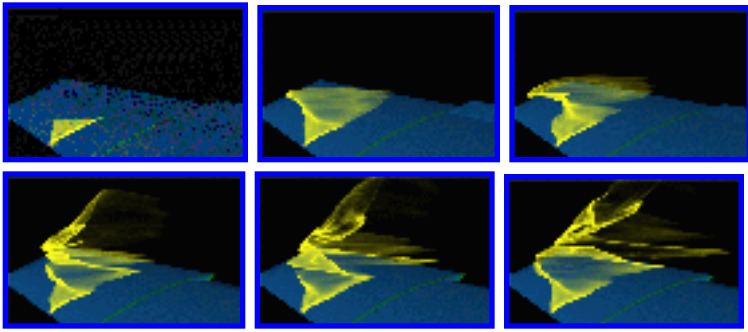
High-performance Computing Needed

Regardless of the mode of operation, the use of high-performance computing is necessary to improve visualization performance. A common approach is the use of distributed processing, combining the capabilities of a high-performance graphics workstation and a supercomputer under a single application. Supercomputers are especially suited for processing large amounts of data, and provide large disk storage and fast access to the data, whereas workstations are suited for quickly transforming data into images.

Unsteady Visualization a Challenge

Simulations will continue to become more advanced as technology for high performance computing progresses. Since visualization is an essential tool to analyze the large amount of information produced by these simulations, research in unsteady visualization must evolve along with simulation technology. Here's more information on [visualization activities at NAS](#).

[Next Article](#)[Contents](#)[Main Menu](#)[NAS Home](#)

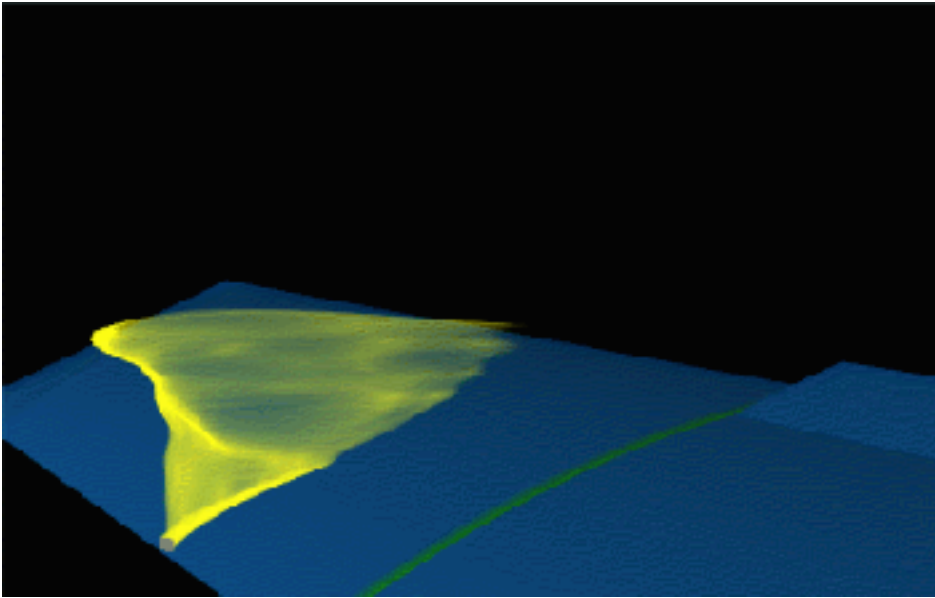


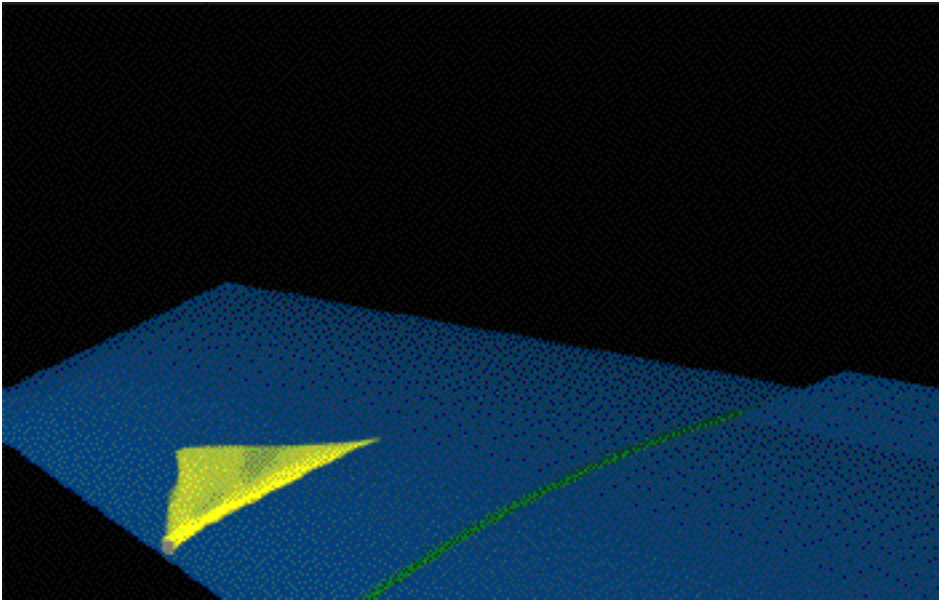
A time series illustrating smoke flowing over a delta wing with oscillating flaps. The smoke is released near the surface of the delta wing and then tracked for three cycles of the flaps. Over time, the smoke dissipates where the flow diverges and becomes dense where it converges.

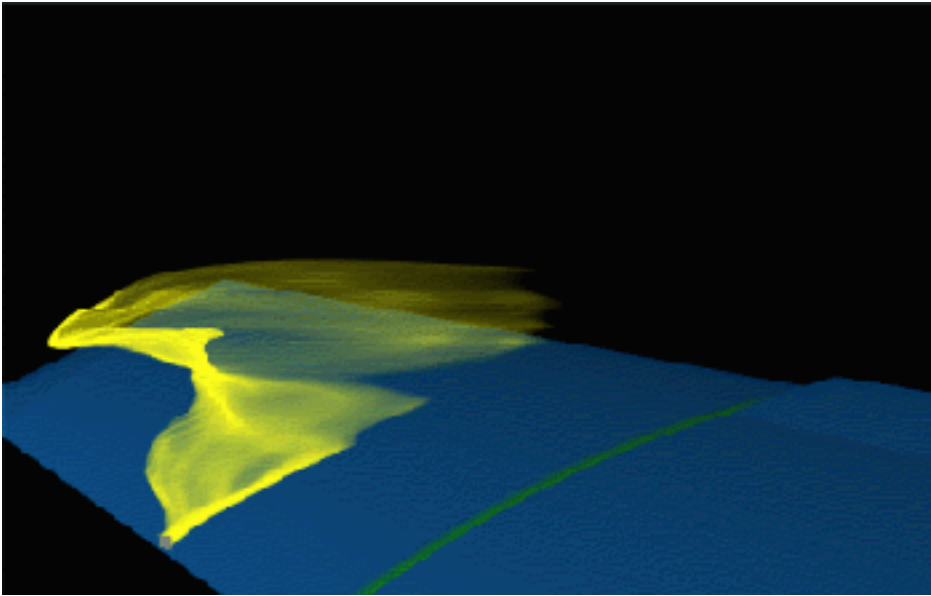
At the early stage, the smoke sweeps over the delta wing surface, then becomes convoluted as it reaches the top of the wing. Later, the smoke twists upward near the wing tip vortex. (Computation of the delta wing by Shigeru Obayashi and Goetz Klopfer.)

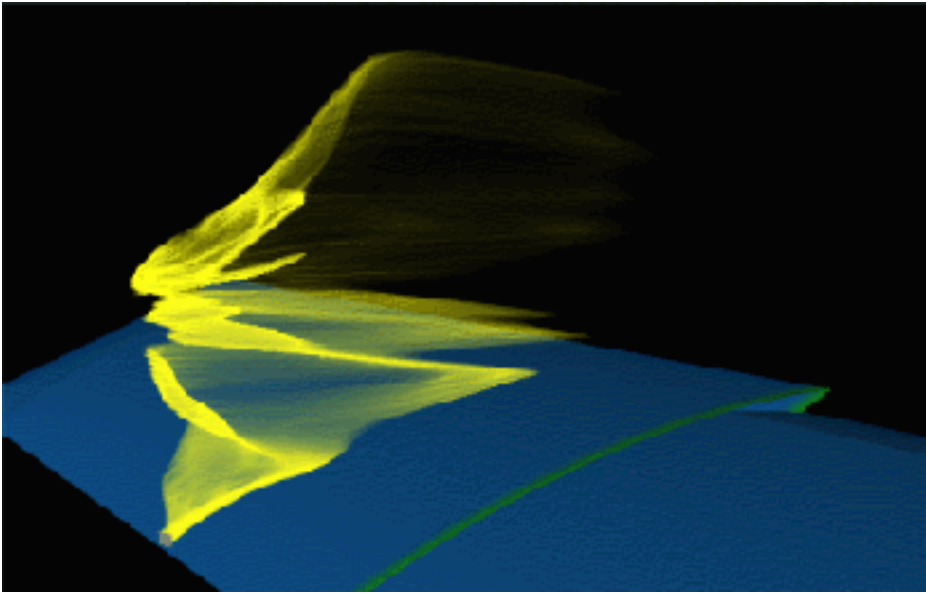


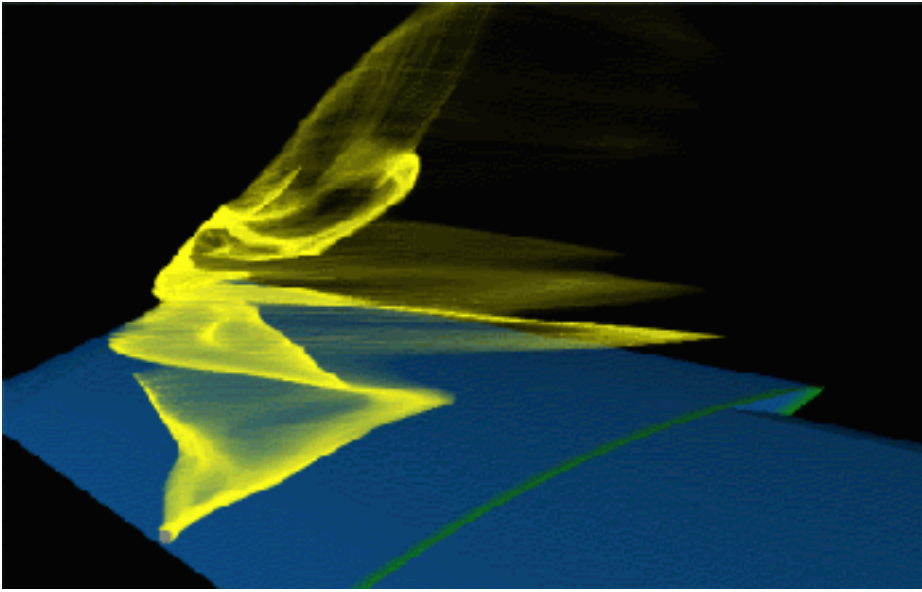
[to the article.](#)

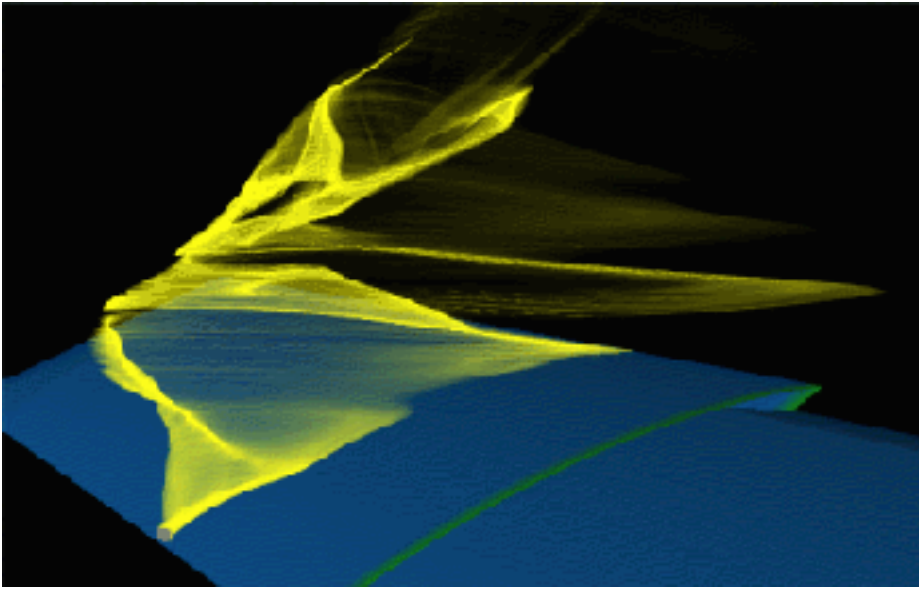








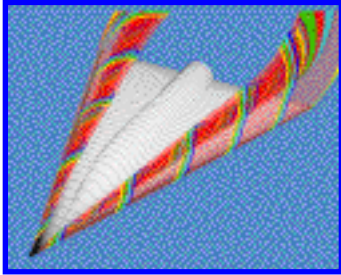




[Next Article](#)[Contents](#)[Main Menu](#)[NAS Home](#)

Improved Techniques for Unsteady Visualization Analysis

By [Elisabeth Wechsler](#)



Scientists want better understanding of their flow simulations offered by unsteady, or "time-dependent," visualization techniques but up to now have been unable to work interactively with their datasets. Particle animations are one of the techniques used to study new aircraft designs. Presently, particle animations are generated in a batch process, sometimes requiring hours to produce. Recent work by David Kenwright, of the NAS visualization technologies group, has contributed a promising solution to this problem.

Kenwright has developed a new time-dependent particle tracer for UFAT [Unsteady Flow Analysis Toolkit], which will reduce calculation times by up to a factor of six. This means that interactive techniques, which are standard in visualization programs such as FAST [Flow Analysis Software Toolkit], will soon be available for unsteady flows.

Unsteady Flow Datasets -- Unwieldy

"Scientists create datasets that describe the airflow around an aircraft, and our group's job is to develop tools that will help them visualize it," Kenwright said. "We haven't been able to offer interactive visualization with unsteady flows because the datasets are on the order of hundreds of gigabytes each, and the algorithms for visualizing unsteady flows are much more complicated than those previously developed for steady flows," he said, adding that this combination requires "considerably more computation time."

Kenwright developed more efficient algorithms for tracing particles in unsteady flows by first analyzing the original algorithms to determine what areas were considered expensive to the process. Two areas became the focal point of Kenwright's work: point location and velocity interpolation.

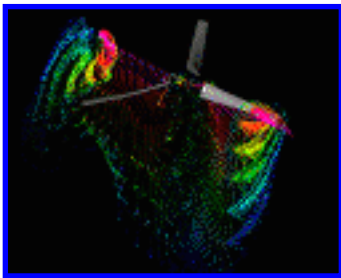
Finding Exactly Where You Are

"Point location is the process of finding exactly where you are in the huge computational grid of an unsteady flow, given an initial spatial point," Kenwright explained. "Velocity interpolation is used to find the value of the velocity field at that point."

Previously, the algorithms for doing point location and velocity interpolation were based on hexahedrons -- essentially, warped cubes, Kenwright said. "I found that by splitting these hexahedrons into tetrahedrons (three dimensional triangular elements), the computations became greatly simplified -- sometimes by as much as six times. Furthermore, tests have shown that the new techniques produce almost identical particle traces to previous algorithms." Eventually, this new particle-tracing technique will be included in future products.

Adding Interactive Controls

One benefit of Kenwright's breakthrough is that "calculations have been brought down to a level where the scientist will be able to do more interactively, such as changing the locations for releasing particles, which will allow them to find important features in their flows," Kenwright said, adding that interactive controls will be available to users in the future. In addition, these algorithms will benefit UFAT users by reducing the time required to generate animations or videos, he said. The [latest release of UFAT](#) includes the new algorithms.



Currently, Kenwright is working with David Lane, who developed UFAT, to implement an algorithm to create stream surfaces (generated by tracing rows of particles), which are "easier to visualize than individual particles or lines." He will adapt an algorithm developed by Jeff Hultquist, formerly at NAS, to create stream surfaces in unsteady flows.

"This technique reveals information not easily visualized with particle traces because lighting and shading can be applied to the surfaces to provide additional visual clues," Kenwright said. "These stream surfaces will be more like the smoke streams researchers study in wind tunnel experiments." [See ["Wind Tunnel Smoke Simulated in Numerical Flow Visualization."](#)]

Problem of Dissipating Smoke

He commented that "a problem with wind tunnels is that the smoke gets dissipated easily and is therefore hard to see. On a computer, you can control the effects of dissipation to give clearer representation of the flow."

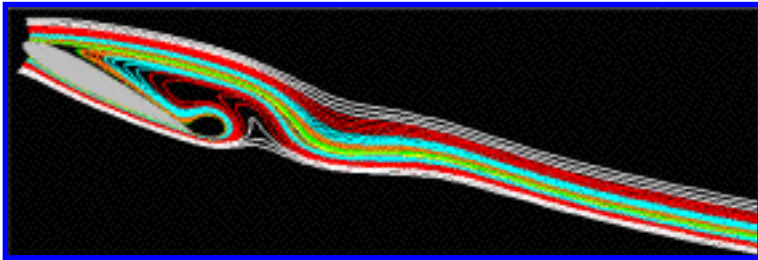
Kenwright and Lane will present a paper on this research at IEEE Visualization '95, Atlanta, in November. For more information about this work, send email to davidk@nas.nasa.gov or lane@nas.nasa.gov.

[Next Article](#)[Contents](#)[Main Menu](#)[NAS Home](#)

[Next Article](#)[Contents](#)[Main Menu](#)[NAS Home](#)

Wind Tunnel Smoke Simulated in Numerical Flow Visualization

By [Elisabeth Wechsler](#)



The study of unsteady flow, traditionally done by releasing smoke on an aircraft model in a wind tunnel -- or, alternatively, dye in water tank experiments -- hasn't translated well for computer simulations up to now. However, a collaboration in progress between NAS scientist David Lane and

counterparts at Lawrence Livermore National Laboratory (LLNL) has created a successful smoke test prototype that runs on a Silicon Graphics Inc. workstation using IRIS Explorer visualization software (now supported by Numerical Algorithms Group).

Adapting previous work on steady flow volumes by Barry Becker and Nelson Max, of LLNL, Lane and Becker have enhanced the software to compute unsteady flow volumes with multizoned curvilinear grids, using some elements of UFAT [Unsteady Flow Analysis Toolkit], which Lane developed.

Problem With Unsteady Flow

One problem with trying to convert steady flow volume into unsteady flow volume is that the latter is based on streaklines and can distort more radically than the streamlines on which a steady flow volume is based. "Streaklines are more dynamic than streamlines -- therefore, they reveal more time-varying phenomena in the flow when animated," Lane explained.

"Smoke is constructed by advecting the vertices of tetrahedrons released from an initial polygon. The vertices can become separated in time and the tetrahedrons become stretched," he continued. "To emulate real smoke and reduce artifacts, the tetrahedrons become more transparent if their volume increases. This gives the appearance of smoke dissipating over time."

System Calls to UFAT Library

Another challenge is extending the method to multizoned curvilinear grids such as those used in the space shuttle, because the existing system for steady flow volume can only handle single-zoned curvilinear grids, Lane said. The new system takes advantage of UFAT's particle-tracing library in time-

dependent flow fields by making calls to UFAT, which includes the particle advection library for multizoned curvilinear grids with motion. These features will become more important as unsteady grids become increasingly popular for use in numerical flow simulations, he added.

Lane can demonstrate the prototype to visitors at his office at NASA Ames, and plans to get feedback from CFD researchers to determine their level of interest in using such a system.

Collaborators 'Happy With Progress'

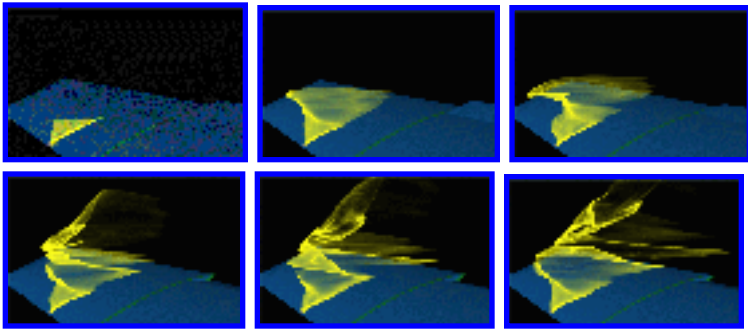
Lane and Becker met at Visualization '93, where they discovered a mutual interest in finding a workaround for smoke tests in computer simulations. Since then, they've communicated regularly and meet at least once a month. The actual collaboration began a year ago and, so far, the two scientists are "very happy with their progress to date," according to Lane. The two researchers will present a paper detailing their interim results at IEEE Visualization '95, Atlanta, in November. LLNL will maintain ownership and release rights for the prototype, but plans to release the software in the public domain.

For more information, send email to lane@nas.nasa.gov or becker1@llnl.gov.

[David Lane](#) is currently investigating surface flow oil visualization and other techniques for UFAT [Unsteady Flow Analysis Toolkit], which he developed at NAS. He earned his doctorate in computer science at Arizona State University, where he developed several software applications for data modeling, computer graphics, and scientific visualization. Lane, who immigrated to the U.S. with his family at 13, likes "the research-and-development atmosphere at NAS" and the "opportunity to be among [those] who solve large-scale scientific visualization problems with state-of-the-art technologies." He has been at NAS for four years.



[Next Article](#)[Contents](#)[Main Menu](#)[NAS Home](#)



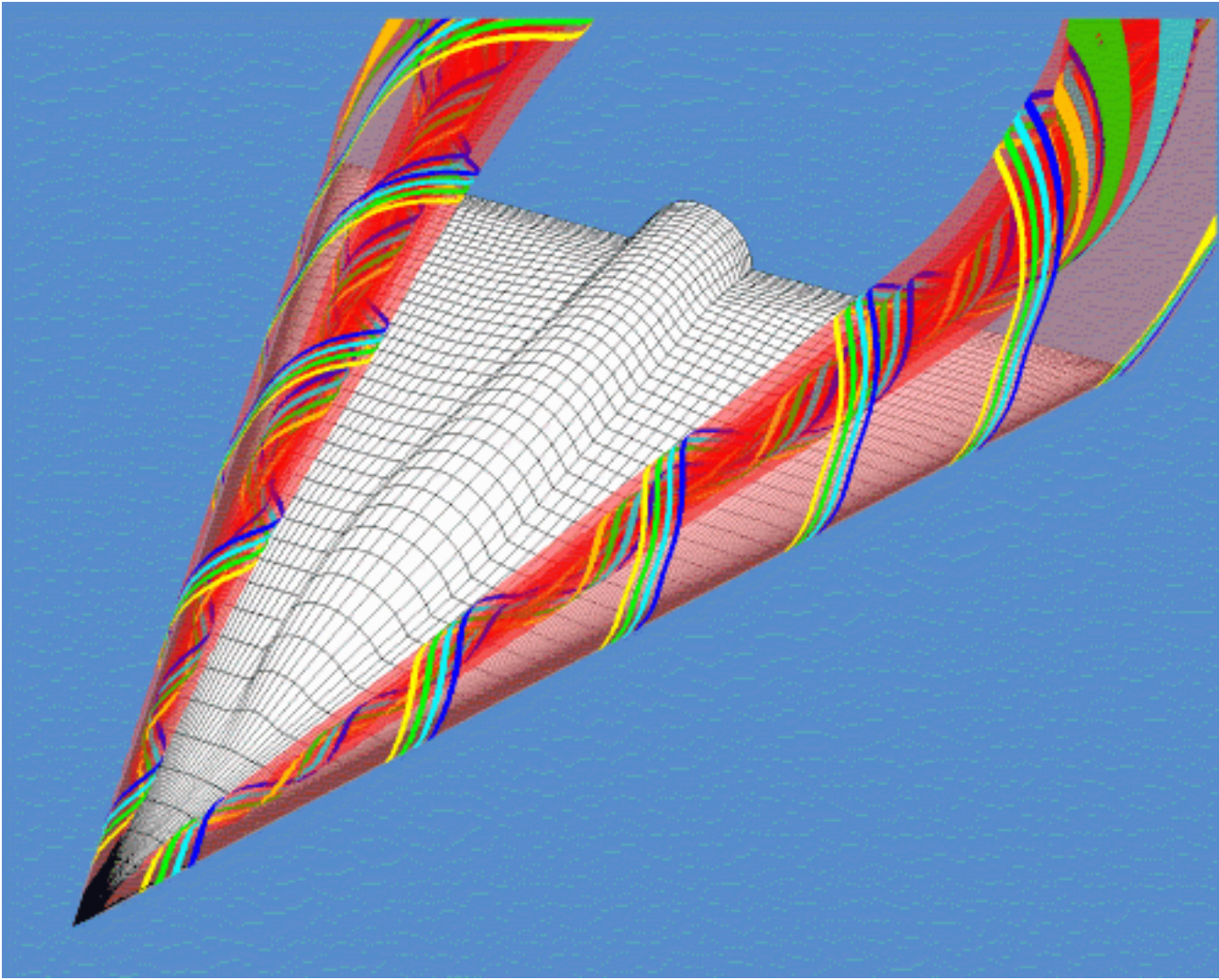
[Mpeg view \(193667 bytes\)](#)

A time series illustrating smoke flowing over a delta wing with oscillating flaps. The smoke is released near the surface of the delta wing and then tracked for three cycles of the flaps. Over time, the smoke dissipates where the flow diverges and becomes dense where it converges.

At the early stage, the smoke sweeps over the delta wing surface, then becomes convoluted as it reaches the top of the wing. Later, the smoke twists upward near the wing tip vortex. (Computation of the delta wing by Shigeru Obayashi and Goetz Klopfer.)



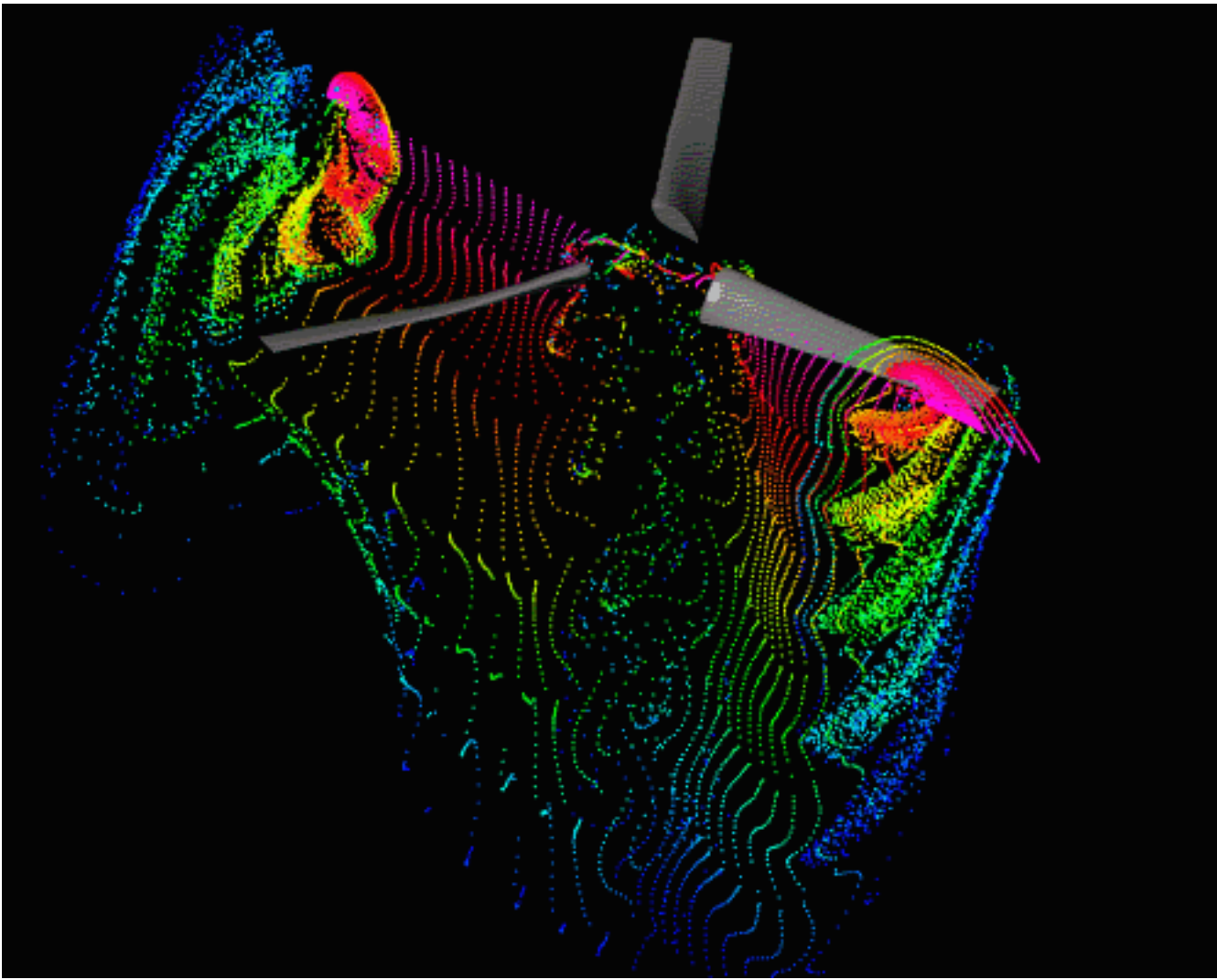
[to the article.](#)



Stream surfaces and stream ribbons provide insight into three-dimensional flow features, which are difficult to visualize with individual particles. An example is the vortex roll-up across the leading edge of this delta wing. Transparency and lighting are used to highlight the obscured surfaces in this visualization. (Graphics by Jeff Hultquist; flow computations by Neal Chaderjian.)



[to the article.](#)



Particle animations in time-dependent flows are analogous to smoke streams in wind tunnel experiments. They convey information to scientists about the motion and dynamics of a flow. In this visualization, particle paths were computed using UFAT, the Unsteady Flow Analysis Toolkit, from a flow simulation of a V-22 tiltrotor aircraft. (Graphics by David Kenwright; flow computations by Robert Meakin.)



[to the article.](#)

[Next Article](#)[Contents](#)[Main Menu](#)[NAS Home](#)

pV3 Combines Flow Solver, Parallel Interactive Viewing, Feature Extraction

By [Elisabeth Wechsler](#)

pV3 [parallel Visual 3] provides a parallel interactive graphics environment for visualizing three-dimensional (3D), unstructured and/or structured volumetric data, which may be steady or unsteady. This tool is designed to run in parallel on a network of distributed workstations. pV3 and its precursor, Visual 3 (designed to run on a single 3D workstation), were developed by Bob Haimes of Massachusetts Institute of Technology's Department of Aeronautics and Astronautics. Haimes received partial funding for his work from NASA's Ames and Lewis Research Centers.

Timothy Barth, a CFD researcher at Ames has used pV3 on the NAS IBM SP2 [*see [NAS News, May-June '95](#)*] and appreciates the way the tool efficiently handles "very general element shapes, such as hexahedrons, tetrahedrons, prisms, and pyramids." pV3 includes many of the traditional visualization tools of FAST and PLOT3D, he noted.

Examines Solution Subsets

"This tool allows researchers to look at a subset of a mesh and numerical solution, which is difficult to do with an unstructured grid," Barth said. Also, pV3 "lets you produce the mesh in subdomains [a situation typical in CFD parallel computation algorithms] and the program will put them together automatically," Haimes added.

"If you have an unsteady problem, you must dump an incredible amount of data on disk and then do postprocessing visualization," Haimes said. "Because the event is already done, you can't affect the solution," he explained, adding that an analogy would be "filming something for later viewing rather than experiencing it now."

With pV3, the solver and visualization system are combined, Haimes said. Previously, as with PLOT3D and FAST, scientists went through three steps to analyze unsteady problems: generating the grid, running a solver, and then conducting postprocessing visualization. "This still works fine for steady state problems," he added. However, "for CFD and transient problems, the difficulty is dealing with time and vast amounts of data."

Animations Don't Help

"Except for UFAT [Unsteady Flow Analysis Toolkit], the conventional approach to visualize unsteady data is to take snapshots in time and then treat the data like a steady-state problem by creating an animation and playing it back like a movie," Haimes said. "While this form is great for communicating ideas, it stinks if you're trying to understand the physics in the data," he said. "If something is wrong with your design, or if you're trying to find some salient feature, it's difficult with this method."

Not being able to see anything but what has been "story-boarded," and the problem of too much data requiring large amounts of disk space, are two problems with the traditional approach. Realizing that "there had to be a better way," Haimes designed pV3 around the [concept of "coprocessing,"](#) or the concurrent execution of the flow solver with the visualization system.

"You can watch the solution as things change. You can pause it and look around, plus there is the added benefit of being able to steer the solution," Haimes said. To do this, you can "change some boundary condition of your solver -- more trivial -- or change the model physically and [watch the results propagate through the flow,](#)" he continued. The latter form of steering lets researchers verify that a design has the characteristics to function correctly.

"One caveat to this approach is that large-scale CFD computations, such as the V-22 aircraft, on conventional vector supercomputers can require several minutes for each time step," Barth said, acknowledging that this delay might make pV3 a less attractive option in this situation. He added, however, that the number-crunching portion of the total time would be "substantially reduced on parallel supercomputers, such as the SP2."

Debugs Steady-state Algorithms

Barth uses pV3 to analyze steady-state problems so that he can watch the visualization of a solution -- not just the numbers -- as it converges. This helps him debug algorithms. Also, by examining his data with pV3 on the SP2, he can view data interface regions and discern whether information transferred between regions is correct.

"The main benefits of pV3 are being able to distribute problems across several workstations or execute the pV3 library on a real parallel machine like the SP2," Barth said. "Before, you could only run small problems on a workstation and it was slow. Now, the SP2 can handle jobs 200 times larger than a simple workstation's capacity."

In addition, Barth commented that "pV3 lets you extract data and study it in both space and time. It brings back the minimum amount of information to render an image or flow quantity. Otherwise, it would easily overwhelm the network. This way, it reduces the problem to something more manageable." He added that pV3 "updates simultaneously the parallel computation as well as the slice you've extracted to view on a graphics workstation."

Supports PVM and MPI Protocols

The tool exploits embarrassingly parallel computations, typical in flow visualization graphics, so that "when run on the SP2 you get parallel speedup with each processor doing a piece of the mesh," Barth said. pV3 uses PVM [Parallel Virtual Machine] message-passing protocol, in which the "message passing is hidden and otherwise transparent to the pV3 user. In fact, on the SP2 it's possible to use MPI [Message Passing Interface] protocol in a flow solver code and still use pV3," he added.

Haimes pointed out that using, for example, 64 nodes of the SP2 allows calculations of millions of points: "If the results were put together in one file for later viewing, it could require more memory than you have available on any single workstation."

Haimes acknowledged that pV3 is "not easy to use. There's a learning curve -- you either have to be adventurous and curious, or you have to be under the gun" to try it. However, Barth commented that "you need to know very little about what pV3 is doing -- it's clean and non-ambiguous."

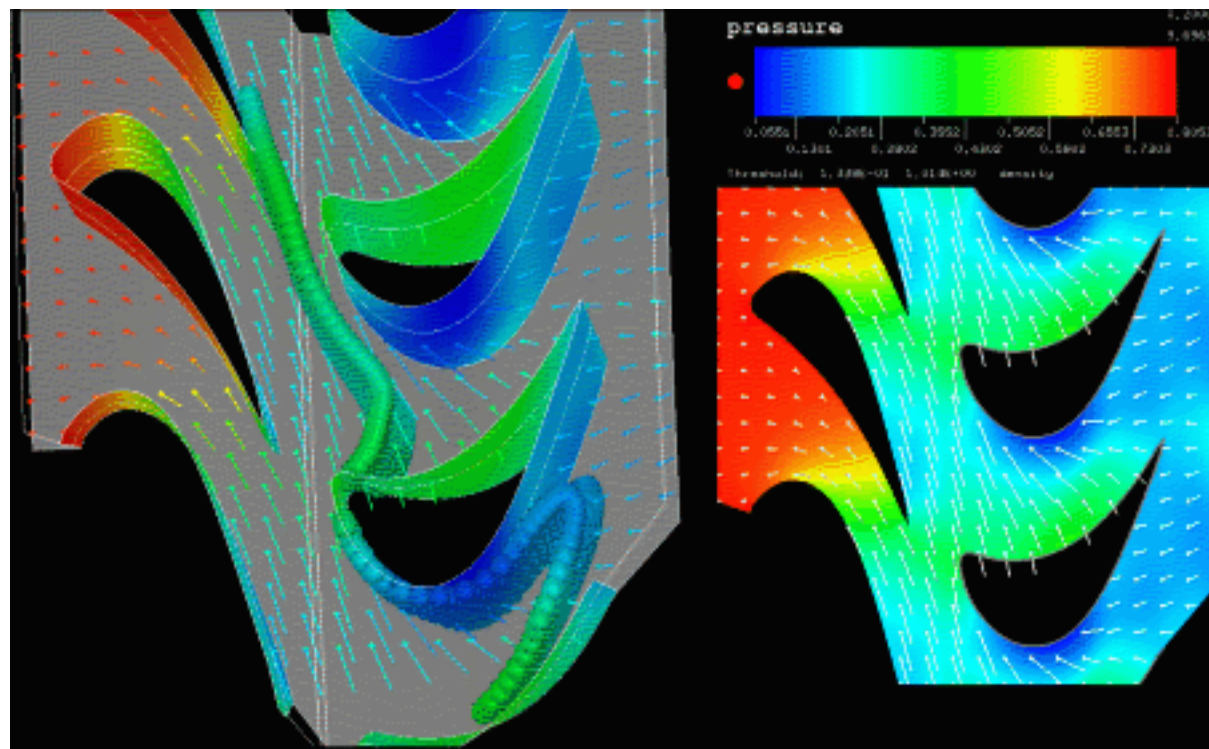
Improved Extraction Techniques

In the months ahead, Haimes plans to add to, as well as refine, pV3's feature extraction techniques. "We need a way of gleaning all this information without using traditional postprocessing methods. You can spend a lot of time on one snapshot of a dataset, but there may be thousands," he said.

An example of one successful application of feature extraction with pV3 is described in a recent paper, entitled "Identification of Swirling Flow in 3-D Vector Fields," that Haimes (with co-author David Sujudi, of MIT) presented at the AIAA CFD meeting, San Diego, in June.

Here's more information on [pV3](#). (including technical documentation as well as software distribution.)

[Next Article](#)[Contents](#)[Main Menu](#)[NAS Home](#)



The pV3 screen during a coprocessing visualization session showing unsteady flow through the first turbine stage of a jet engine. The solver is executing on two different workstations and the result is displayed on a third. A 3D picture of the stator (stationary blade row) and rotor (moving blade row) is shown above, left. The colors denote pressure on the blades while the tufts and streaklines indicate the direction of the flow. The 2D and 1D windows shown below display information extracted from the 3D flow.

In Flow ☐ ☐ ☐ ☐

Out Flow ☐ ☐ ☐ ☐

Hub ☒ ☐ ☐ ☐

Tip ☐ ☐ ☐ ☐

Stators ☒ ☐ ☐ ☐

Rotors ☒ ☐ ☐ ☐

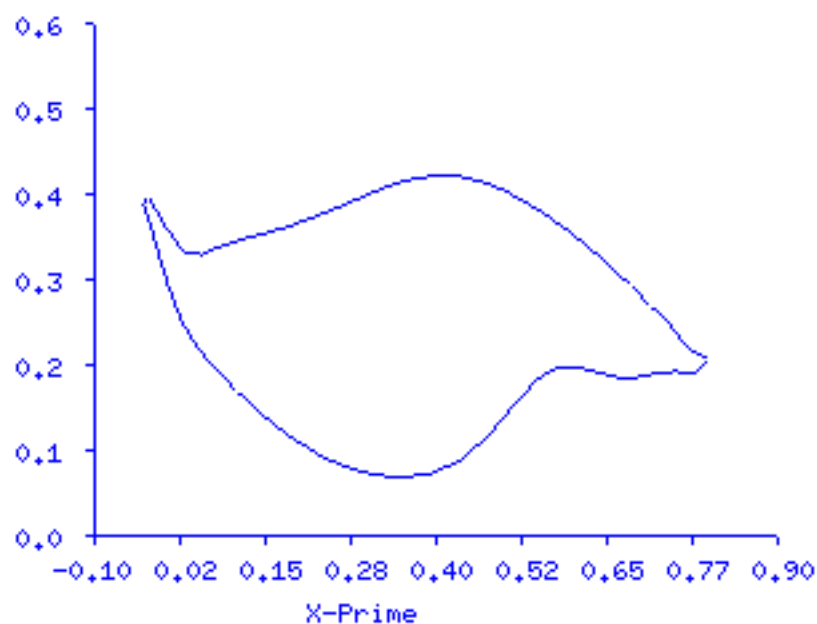
Others ☐ ☐ ☐ ☐

Render/Grid/Grey/Thresh

Surfaces

Unsteady Stator/Rotor VISUAL3 Demo

pressure

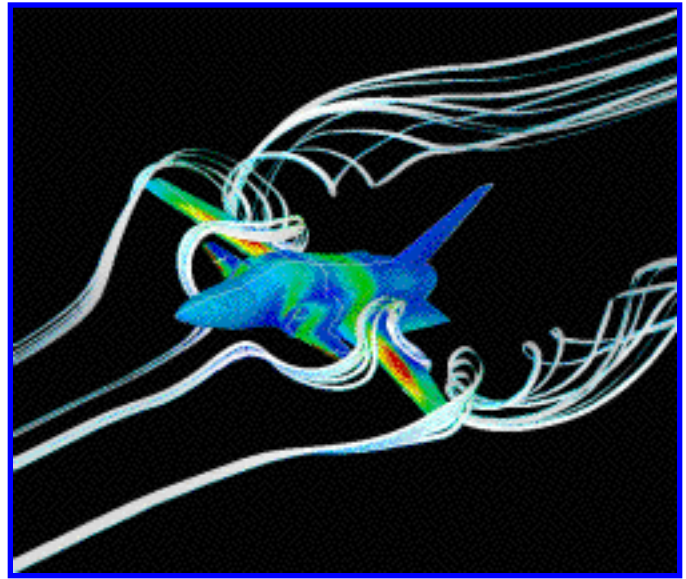


Period = 0.0000

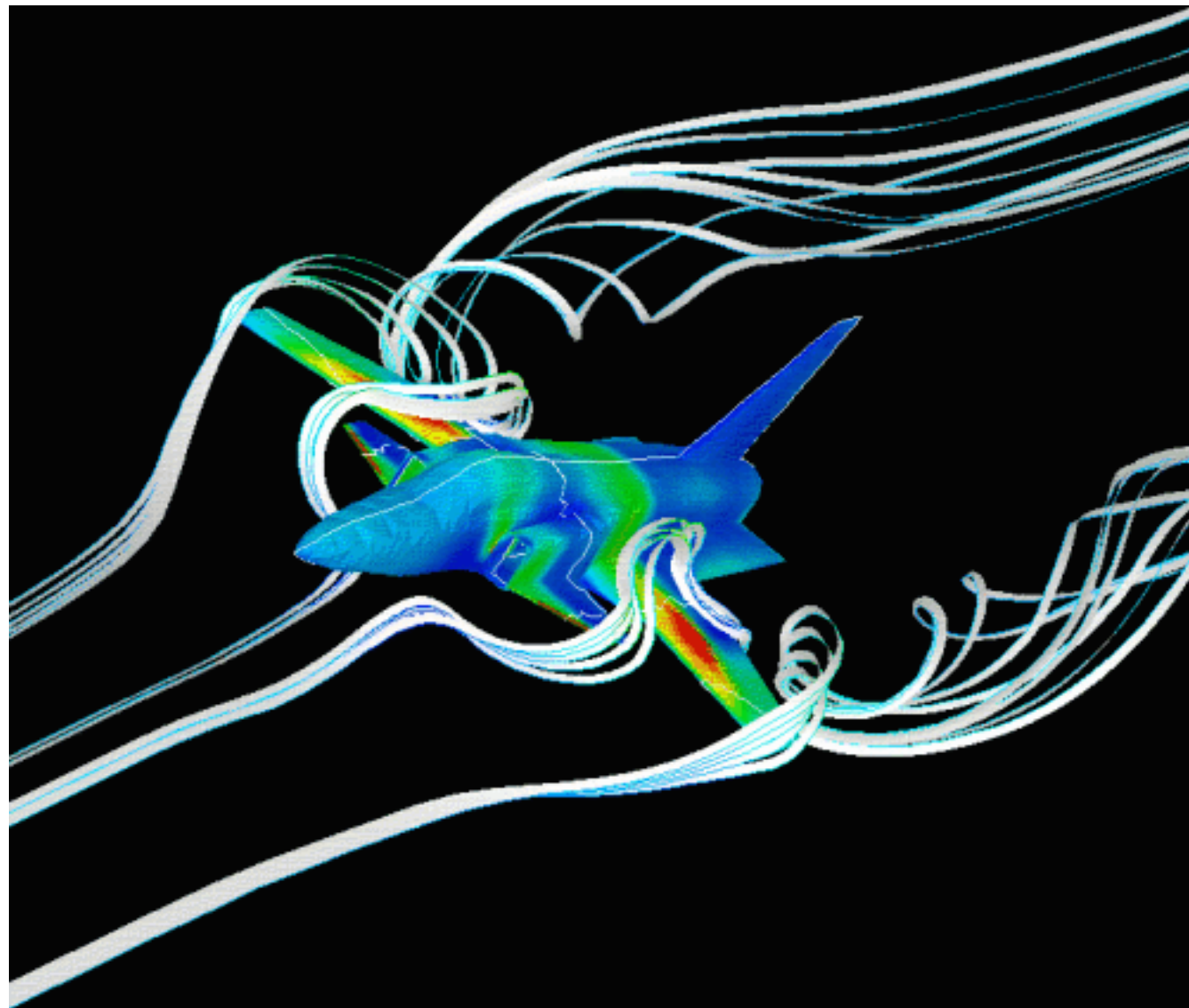


[to the article.](#)

This image displays an X29 aircraft flying at 8 degrees' angle of attack. The domain has been partitioned into four subdomains, and the solution was computed on the NAS IBM SP2. The graphic displays mach number on the surface of the aircraft. Two sets of instantaneous stream ribbons are shown (one set at the tip of the canard, the other at the tip of the wing). The unusual flow features have been found by using parallel Visual 3's steering capability. The calculation was performed by pFELISA (the parallel version of FELISA, a CFD system for unstructured grid generation and solving; the code was written by Jaime Peraire, of MIT).



[to the article.](#)



[Next Article](#)[Contents](#)[Main Menu](#)[NAS Home](#)

Unsteady Flow Technique Reveals New Phenomena

By [Elisabeth Wechsler](#)

While implementing unsteady particle tracing algorithms, UFAT [Unsteady Flow Analysis Toolkit] developer David Lane, of the NAS visualization technologies group, has investigated other unsteady flow visualization techniques and compared them with existing steady methods.

Techniques such as streamlines, contour lines, arrow plots, and cutting planes are standard for visualizing steady flows. However, most of these techniques do not adequately depict time-varying phenomena (for example, vortex shedding and vortex breakdown) in unsteady flows because time was not factored into the calculation, Lane explained. He added that particle tracing is an effective technique for visualizing flow directions and has been widely used for steady flows.

Lane has adapted particle-tracing techniques for unsteady flows with multizoned curvilinear moving grids. This work involved developing time dependent algorithms in particle tracing -- for example, particle integration, velocity interpolation, and grid interpolation, in both time and space.

Steady vs. Unsteady Flow Techniques

Lane [compared several visualization techniques](#) and found the information revealed by steady and unsteady flows to be very different. "Because steady techniques only use data from the flow at one instant in time, these techniques are sometimes referred to as instantaneous flow visualization techniques and are ideal for analyzing steady flows," he said.

However, in unsteady flows, where flow quantities vary in time, steady techniques may not capture the time evolution of flow features, Lane continued. "Because unsteady techniques are based on many time steps of the flow data, they can capture time-varying flow features effectively," he explained.

Lane plans to submit a video of his comparison results to the Visualizing Time-Varying Data Symposium, to be held in Williamsburg, VA, in September. The symposium is sponsored by the Institute for Computer Applications in Science and Engineering and NASA Langley Research Center in cooperation with ACM SIGGRAPH. This work will also be presented in a paper entitled, "Visualizing Time-Varying Phenomena in Numerical Simulations of Unsteady Flows," which Lane has submitted to the AIAA Aerospace Sciences Meeting, to be held in Reno next January.

Focus on Surface Oil Flows

Another area of Lane's focus is the surface oil flow technique for unsteady flows. Responding to requests from CFD researchers at NASA Ames, he is investigating the use of the Line Integral Convolution (LIC) technique for unsteady flows, which was introduced at SIGGRAPH '93 by Brian Cabral and Casey Leedom (formerly of Lawrence Livermore National Laboratory and now at Silicon Graphics Inc.) The technique was then extended to curvilinear grids by Lisa Forssell (see [NAS News January-February '95](#)).

"Traditional flow techniques simulate surface flows by releasing particles from some discrete grid locations and then tracking them on the grid surface," Lane said. "By watching the surface flow over the grid surface, a scientist hopes to see separation lines and flow attachments."

He explained that in order to visualize these flow features, careful placement of the release points for particle traces is required. "The LIC technique avoids this problem totally because it reveals the flow pattern on the entire grid surface." Lane will report the results of this work in a future issue of *NAS News*.

For more information, send email to lane@nas.nasa.gov.

[Next Article](#)[Contents](#)[Main Menu](#)[NAS Home](#)

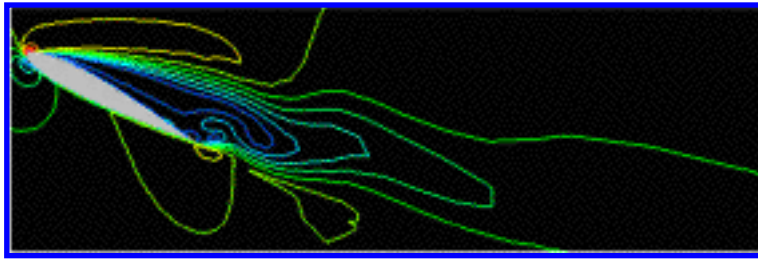


Figure 1

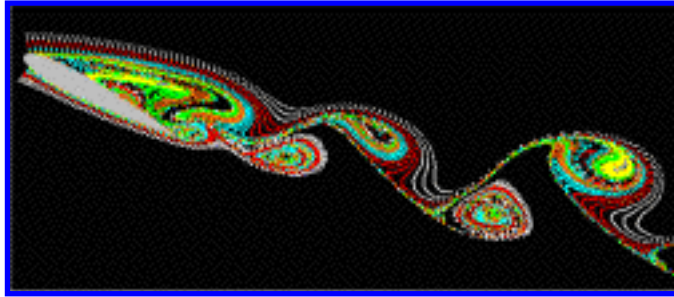


Figure 2

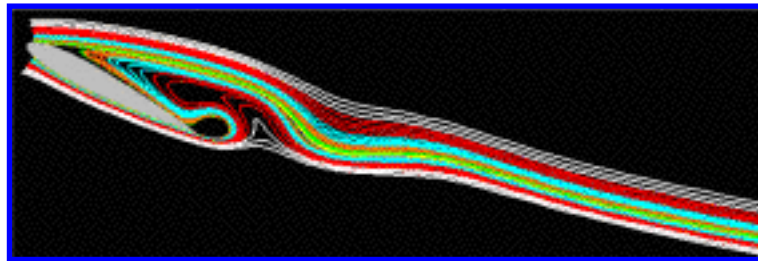


Figure 3

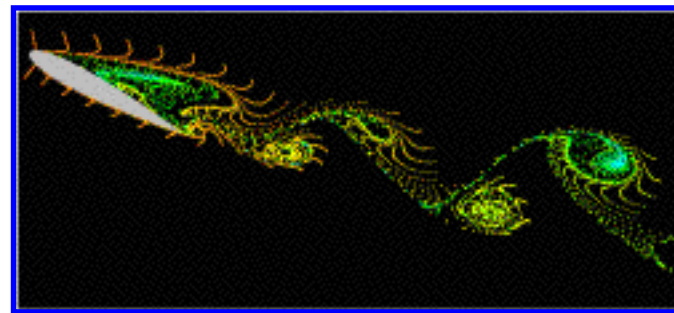


Figure 4

Two steady flow visualization techniques, contour lines ([Figure 1](#)) and streamlines ([Figure 2](#)) are compared with two unsteady flow techniques, streaklines ([Figure 3](#)) and timelines ([Figure 4](#)). The unsteady techniques reveal the vortex structures (circular, disklike shapes) more clearly than do the steady techniques. Contour lines and streamlines show the general structure of the vortices, while streaklines and timelines give more detail. (Flow computations by Sungho Ko.)



[to the article.](#)

